International Journal of Computer Science Engineering
and Information Technology Research (IJCSEITR)
ISSN(P): 2249-6831; ISSN(E): 2249-7943
Vol. 3, Issue 5, Dec 2013, 103-114
© TJPRC Pvt. Ltd.

ENABLING, EFFICIENT RANKED KEYWORD SEARCH, ENCRYPTED FILES AND UPLOADING ON OUTSOURCED CLOUD DATA

DOAA MOHSIN MAJEED[1], W. JEBERSON[2] & I. B. RAJWADE[3]

[1]M.Sc.Computer Science, Republic of Iraq, Sam Higginbottom Institute of Agriculture Technology & Sciences, Allahabad, Uttar Pradesh, India

[2]Associate Professor, Department of Computer Science & I.T, Sam Higginbottom Institute of Agriculture Technology & Sciences, Allahabad, Uttar Pradesh, India

[3]Assistant Professor, Department of Computer Scinece & I.T, Sam Higginbottom Institute of Agriculture Technology & Sciences, Allahabad, Uttar Pradesh, India

ABSTRACT

Cloud computing has become an integral part of IT industry. Amount of information available on World Wide Web is increasing at an exponential pace. In such a vast collection it becomes difficult for the user to query something out of the whole collection. Great efforts have been made for facilitating users via keyword search. However, there are a few researches about entertaining the exact user query and presenting a ranked URL list according to it. In this project, We give an overview of our framework for keyword searching with summaries, besides we describe a ranking algorithm for ranked keyword search and their results. Keyword searches are typically done so that users can actively utilize clouds to query a collection.

Search engine companies collect the "database of intentions", the histories of their users' search queries. To protect data privacy, sensitive cloud data have to be encrypted before outsourced to the commercial public cloud, which makes effective data deployment service a very challenging task. These search logs are a gold mine for researchers. Search engine companies, however, are wary of publishing search logs in order not to disclose sensitive information. This report present analyzes algorithms for publishing frequent keywords, queries and clicks of a search log.

Ranked search greatly enhances system usability by enabling search result relevance ranking instead of sending un-differentiated results, and further ensures the file retrieval accuracy. The resulting design is able to facilitate efficient server-side ranking without losing keyword privacy. Then demonstrate that the stronger guarantee ensured by differential privacy unfortunately does not provide any utility for this problem.

KEYWORDS: Data Oriented Cloud, Data Outsourcing, Privacy Based Aspect, Data Security, Mapping of Preserved Order, Encryption of the Searchable Data, Search Based Ranking Strategy and Index Search Respectively

INTRODUCTION

Cloud computing is the delivery of computing services over the Internet. Cloud services allow individuals and businesses to use software and hardware that are managed by third parties at remote locations. Examples of cloud services include online file storage, social networking sites, webmail, and online business applications. The cloud computing model allows access to information and computer resources from anywhere that a network connection is available. Cloud computing provides a shared pool of resources, including data storage space, networks, computer processing power, and specialized corporate and user applications .Our application performs the following operations:

- User can register as Data Owner (the user who can provide the cloud information).
- User can register as Data User (the user who can search the information).
- Admin, Data Owner & Data User Login Interfaces & Login Operations
- Data Owner can upload file to the cloud.
- Due to security reasons uploaded data will be encrypted and create an index for keyword searching.
- After search result if result is found then the user can download the data and it will be automatically decrypted into the actual format.
- Data Admin can clear all the information & can update the database information.

MODULE DESCRIPTION

The following modules are introduced in to our application.

- Encrypt Module
- Client Module
- Multi-keyword Module
- Admin Module
- Uploading / Downloading of files

Encrypt Module

This module is used to help the server to encrypt the document using base64 encoding.

Base64 Encoding

Base64 encoding is reasonably efficient (encoded data is roughly one third larger than the original data), and has the advantage of being highly compatible with most operating systems because it only uses a limited character set in its encoded form.

The algorithm is described as part of the MIME specification. Also, RFC 3548 describes Base64 encoding, but takes a slightly different attitude to line breaks. We will discuss this below.

Algorithm

Base64 encodes the input data three bytes at a time. Each block of three input bytes is encoded to create a block of four printable characters.

The three bytes are ordered into a 24 bit value, starting from the most significant bit of Byte0, and ending with the least significant bit of Byte2. The bits are then arranged as a set of four 6 bit numbers, N0 to N3.as figure 1 The first 6 bits form N0, the next 6 form N1 etc. Each 6 bit number has a range 0 to 63

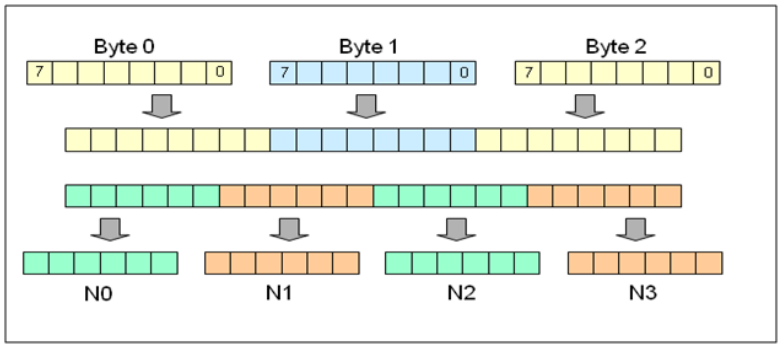

Figure 1: Bit Shift Pattern

The second stage is to convert numbers N0…N3 into ASCII characters, C0…C3. This is done according to the following table (1) below:

Table 1: ASCII Characters

0	A	16	Q	32	g	48	w
1	B	17	R	33	h	49	x
2	C	18	S	34	I	50	y
3	D	19	T	35	j	51	z
4	E	20	U	36	k	52	0
5	F	21	V	37	l	53	1
6	G	22	W	38	m	54	2
7	H	23	X	39	n	55	3
8	I	24	Y	40	o	56	4
9	J	25	Z	41	p	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	t	61	9
14	O	30	e	46	u	62	+
15	P	31	f	47	v	63	/

In addition, the MIME specification states that encoded data should have a CRLF character pair inserted after every 76 characters (or less) of encoded data. The original data can be any length, not necessarily a multiple of 3. This means that the last block of binary data could be 1, 2 or 3 bytes long. To code the final block, we add zeros to the final block to make it a multiple of 3, and convert it to 4 characters as usual. However, we indicate the length of the block in the following way:

- If the final block has a length of 1 byte, the encoded characters consist of C0, C1, followed by 2 "=" characters (C2 and C3 contain no useful information anyway).
- If the final block has a length of 2 bytes, the encoded characters consist of C0, C1, C2, followed by a single "=" character.
- If the final block has a length of 3 bytes, the encoded characters consist of C0, C1, C2, C3 in the normal way.

Client Module

This module is used to help the client to search the file using the multiple key words concept and get the accurate result list based on the user query. The user is going to select the required file and register the user details and get activation code in mail from the "customerservice404" email before enter the activation code. After user can download the Zip file and extract that file.

Multi-Keyword Module

This module is used to help the user to get the accurate result based on the multiple keyword concepts. The users can enter the multiple words query, the server is going to split that query into a single word after search that word file in our database. Finally, display the matched word list from the database and the user gets the file from that list.

Admin Module

This module is used to help the server to view details and upload files with the security. Admin uses the log key to the login time. Before the admin logout, change the log key. The admin can change the password after the login and view the user downloading details and the counting of file request details on flowchart. The admin can upload the file after the conversion of the Zip file format.

Upload & Download Module

Uploading is the transmission of a file from one computer system to another, usually larger computer system. From a network user's point-of-view, to upload a file is to send it to another computer that is set up to receive it. For File upload, we will use Apache Commons File Upload utility, for our project we are using version 1.3, File Upload depends on Apache Commons IO jar, so we need to place both in the lib directory of the project, as you can see that in above image for project structure.

SEARCH MODULE

For searching here we use the hashing technology.

Hashing

Hashing in its simplest form, is a way to assigning a unique code for any variable/object after applying any formula/algorithm on its properties. A true Hashing function must follow this rule:

Hash function should return the same hash code each and every time, when function is applied on same or equal objects. In other words, two equal objects must produce same hash code consistently.

Hash Map

The Map interface maps unique keys to value means it associate value to unique keys which you use to retrieve value at a later date. Some of the key points are:-

Using a key and a value, you can store the value in Map object. You can retrieve it later by using it's key.

- When no element exists in the invoking Map, many methods throw a 'No Such Element Exception'.
- A Class Cast Exception is thrown when an object is incompatible with the elements in a map.
- A Null Pointer Exception is thrown if an attempt is made to use a null object and null is not allowed in the map.
- An Unsupported Operation Exception is thrown when an attempt is made to change an un modifiable map.

INDEXING

Basic Concepts of Index

An index for a file works like a catalogue in a library. Cards in alphabetic order tell us where to find books by a particular author. In real-world databases, indices like this might be too large to be efficient. We'll look at more

sophisticated indexing techniques, There are two kinds of indices.

- Ordered indices: indices are based on a sorted ordering of the values.
- Hash indices: indices are based on the values being distributed uniformly across a range of buckets. The buckets to which a value is assigned is determined by a function, called a hash function.

We will consider several indexing techniques. No one technique is the best. Each technique is best suited for a particular database application. Methods will be evaluated on Access Types of access that are supported efficiently, e.g., value-based search or range search.

- Access Time -- time to find a particular data item or set of items.
- Insertion Time -- time taken to insert a new data item (includes time to find the right place to insert).
- Deletion Time -- time to delete an item (includes time taken to find item, as well as to update the index structure).
- Space Overhead -- additional space occupied by an index structure.

We may have more than one index or hash function for a file. (The library may have card catalogues by author, subject or title.)

The attribute or set of attributes used to look up records in a file is called the search key (not to be confused with primary key, etc.).

Ordered Indices

- In order to allow fast random access, an index structure may be used.
- A file may have several indices on different search keys.
- If the file containing the records is sequentially ordered, the index whose search key specifies the sequential order of the file is the primary index, or clustering index. Note: The search key of a primary index is usually the primary key, but it is not necessarily so.
- Indices whose search key specifies an order different from the sequential order of the file are called the secondary indices, or nonclustering indices.

Primary Index

There are two types of ordered indices:

- **Dense Index**
    - An index record appears for every search key value in file.
    - This record contains search key value and a pointer to the actual record.
- **Sparse Index**
    - Index records are created only for some of the records.
    - To locate a record, we find the index record with the largest search key value less than or equal to the search key value we are looking for.

- We start at that record pointed to by the index record, and proceed along the pointers in the file (that is, sequentially) until we find the desired record.

Figures 2 and 3 show dense and sparse indices for the deposit file.

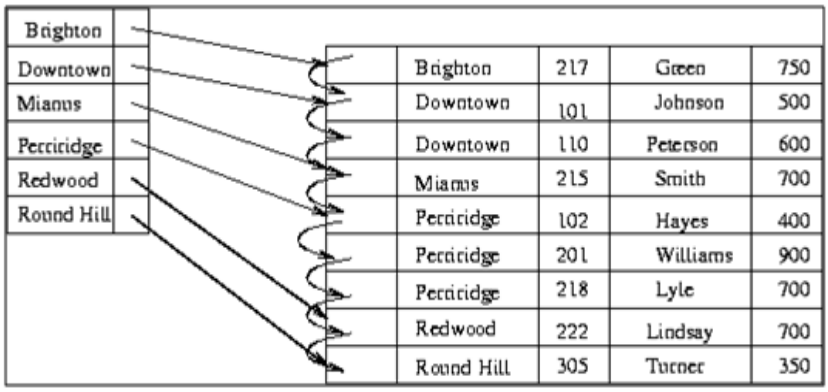

Figure 2: Dense Index

Figure 3: Sparse Index

SYSTEM REQUIREMENTS SPECIFICATION (SRS)

This document aims at defining the overall software requirement for **'**Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data'. Efforts have been made to define the requirements exhaustively and accurately. The final product will be having only features/functionalities mentioned in this document.

In case it is required to have some additional features, a formal change request will need to be raised and subsequently a new release of this document and/or product will be produced.

This specification document describes the capabilities that will be provided by the software application **'**Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data'. It also states the various required constraints by which the system will put up with. The intended audiences for this document are the development team, testing team and end users of the product.

The Scope of Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data is software developed to provide services to the users who would like to get certain information regarding the available cloud data. Also, it would enable the user to perform online registration and to perform several other functions

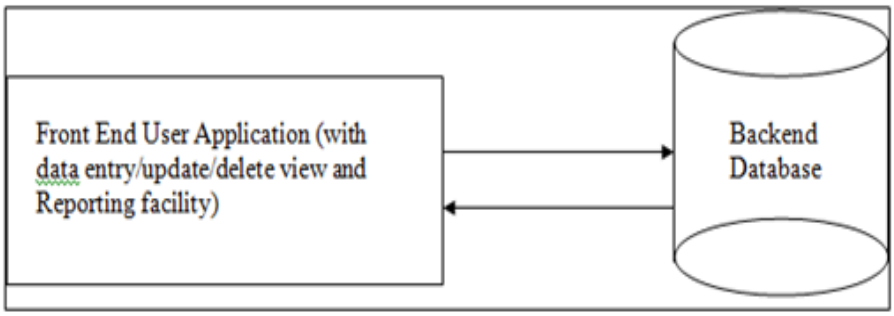

Figure 4: System Database Interaction

HARDWARE REQUIREMENTS

System	:	Pentium IV 2.4 GHz
Hard Disk	:	40 GB
Floppy Drive	:	1.44 MB
Monitor	:	15 VGA color
Mouse	:	Logitech.
Keyboard	:	110 keys enhanced
RAM	:	256 MB

SOFTWARE REQUIREMENTS

O/S	:	Windows XP.
Language	:	Java
Technologies	:	Servlets, jsp, JDK1.6
Data Base	:	SQL Server

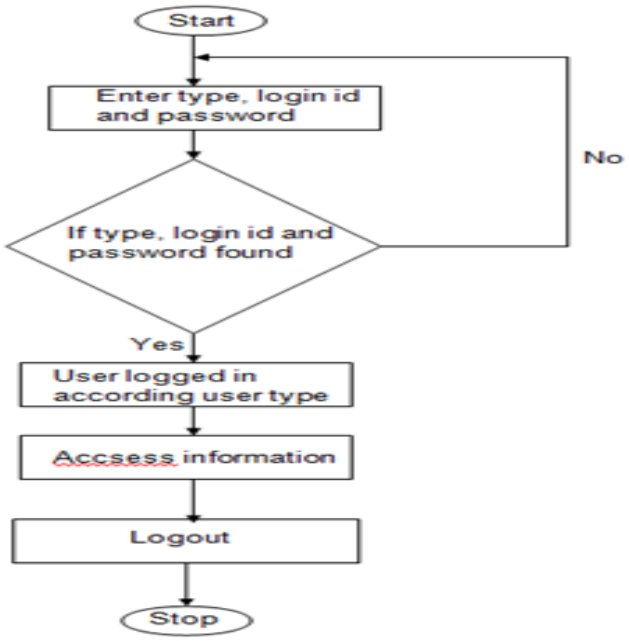

Figure 5: Flow Chart for Login User

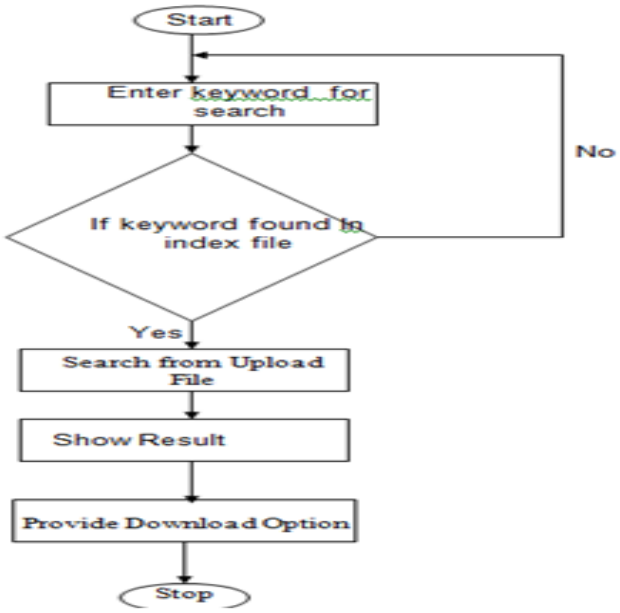

Figure 6: Flow Chart for Search and Download / Administrator

EXECUTION PROCESS

- Owner uploads the file in to the cloud server, and set the privilege to the particular user for easily access data.
- And give the particular permission to download the file with providing the security. Here the user's are separated by authorized user and unauthorized user. Authorized user is the owner permitted person and unauthorized user is unpermitted person
- So authorized user easily access the data from the cloud server by using the ranked efficient keyword search. Unauthorized user can not take the permission to access the data.
- After the data owner permission, then only the authorized user access the data in the cloud server
- The admin can view all user (owner and users) and all details for the owner and user like name , mail id , address, contact number , type of user
- Also the admin has permission to delete any user
- Admin can search about data wise (any file that is downloaded by users at any date and time and recognize the user name and the file is uploaded by which data owner by time also

DISCUSSIONS

- The project is implemented with high security by using the technique of encryption and decryption
- There is efficient searchable is done and so fast ths happened by using index technique.
- The system has been developed by using Java Server Pages for design the interface which a new technology based on java that has many advantages than other programming languages.
- The data has implemented by using MYSQL.
- The system is run as website by using Apcahe Tomcat Server.

CONCLUSIONS

Enabled Secure and Efficient Keyword Search and encrypted the files over Outsourced Cloud Data is a web based project which is being created to help the users, data owners & administrators. Each level of user can generate the enquiry as well as make the updation according to the given privileges. It basically focus on the globalization of the data. A user could see & manage the data as well as an administrator can access the record of an user whenever they required. My project will assist for searching any kind of data using just a keyword and allows for downloading that data whenever is required.

FUTURE SCOPE

I will modify this project for implementing the full text searching as well as different kind of searching technologies like Crawler-based search engine, Human-powered search engine and Hybrid search engine

REFERENCES

1. C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, "Secure ranked keyword search over encrypted cloud data," in Proc. of ICDCS'10, 2010.
2. P. Mell and T. Grance, "Draft nist working definition of cloud computing," Referenced on Jan. 23rd, 2010 Online at http://csrc. nist.gov/groups/SNS/cloud-computing/index.html, 2010.
3. M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica,and M. Zaharia, "Above the clouds: A berkeley view of cloud computing," University of California, Berkeley, Tech. Rep. UCBEECS- 2009-28, Feb 2009.
4. Cloud Security Alliance, "Security guidance for critical areas of focus in cloud computing," 2009, http://www.cloudsecurityalliance.org.
5. Slocum, "Your google docs: Soon in search results?" http:// news.cnet.com/8301-17939 109-10357137-2.html, 2009.
6. Krebs, "Payment Processor Breach May Be Largest Ever," Online at

   http://voices.washingtonpost.com/securityfix/2009/01/ payment processor breach may b.html, Jan. 2009.
7. I. H. Witten, A. Moffat, and T. C. Bell, "Managing gigabytes: Compressing and indexing documents and images," Morgan Kaufmann Publishing, San Francisco, May 1999.
8. D. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in Proc. of IEEE Symposium on Security and Privacy'00, 2000.
9. E.-J. Goh, "Secure indexes," Cryptology ePrint Archive, Report 2003/216, 2003, http://eprint.iacr.org/.
10. D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in Proc. of EUROCRYP'04,volume 3027 of LNCS. Springer, 2004.
11. Y.-C. Chang and M. Mitzenmacher, "Privacy preserving keyword searches on remote encrypted data," in Proc. of ACNS'05, 2005.
12. R. Curtmola, J. A. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: improved," in Proc.of ACM CCS'06, 2006.

13. A. Singhal, “Modern information retrieval: A brief overview,” IEEE Data Engineering Bulletin, vol. 24, no. 4, pp. 35–43, 2001.
14. A. Boldyreva, N. Chenette, Y. Lee, and A. O’Neill, “Orderpreserving symmetric encryption,” in Proc of Eurocrypt’09, volume 5479 of LNCS. Springer, 2009.
15. J. Zobel and A. Moffat, “Exploring the similarity space,” SIGIR Forum, vol. 32, no. 1, pp. 18–34, 1998.
16. O. Goldreich and R. Ostrovsky, “Software protection and simulation on oblivious rams,” Journal of the ACM, vol. 43, no. 3, pp. 431–473, 1996.
17. M. Bellare, A. Boldyreva, and A. O’Neill, “Deterministic and efficiently searchable encryption,” in Proc. of Crypto’07, volume 4622 of LNCS. Springer, 2007.
18. S. Zerr, D. Olmedilla, W. Nejdl, and W. Siberski, “Zerber+r: Top-k retrieval from a confidential index,” in Proc. of EDBT’09, 2009.
19. Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, “Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing,” IEEE Transactions on Parallel and Distributed Systems (TPDS), vol. 22, no. 5, pp. 847–859, May 2011.
20. C. Wang, Q. Wang, K. Ren, and W. Lou, “Towards Secure and Dependable Storage Services in Cloud Computing,” IEEE Transactions on Service Computing (TSC), to appear.
21. http://www.base64decode.org/
22. www.homepages.cwi.nl/~manegold/teaching/DBtech/slides/ch12-8.pdf

AUTHOR’S DETAILS

Doaa Mohsin Majeed was born in Baghdad at June 1988, she received her B.Sc. from university of Kufa ,collage of Educational of girls department of Computer Science at 2011-Ministry of higher education and scientific research , Republic of Iraq , she completed her M Sc. in Computer science form Sam higginbottom Institute of Agriculture Technology & sciences, at 2013 ,Allahabad, U.P, India

Dr. W. Jeberson was born in india he received his B.Sc. physics form Manononmanian sundananar niversity – India at 1996, at 1999 he received M.C.A in computer from Madurai kamraj university – India, 2008 he

received M.B.A in I.T from M.K. University since 2011 he received his phD in computer science from Sam higginbottom Institute of Agriculture Technology & sciences, at ,Allahabad, U.P, India, now he head of Department of computer science Sam higginbottom Institute

Mr. Indresh Bahadur Rajwade born in India he received his B.Sc. P.M. Computer D.A.V.V. Indore at 2001, since 2005 he received M.C.A in computer from AAI-DU-India, now he is assistant prof. in Department of computer science Sam higginbottom Institute of Agriculture Technology & sciences, at, Allahabad, U.P, India,